```
# 1. Install and Load Required Packages
if(!require(biwavelet)) install.packages("biwavelet")
if(!require(tidyverse)) install.packages("tidyverse")
if(!require(readr)) install.packages("readr")
if(!require(tseries)) install.packages("tseries") # For statistical testing
library(biwavelet)
library(tidyverse)
library(readr)
library(tseries)

# 2. Data Import (Please select: "US Full Data.csv")
# A file selection window will appear
csv_path <- file.choose()
# Read CSV
raw_data <- read_csv(csv_path, show_col_types = FALSE)

# Enforce column renaming (ensuring consistency with code variables)
# Expected order: Year, Month, CO2, DOMESTIC, INTERNATIONAL, TOTAL,
EPU
if(ncol(raw_data) >= 7) {
  colnames(raw_data)[1:7] <- c("Year", "Month", "CO2", "DOMESTIC",
"INTERNATIONAL", "TOTAL", "EPU")
} else {
  stop("Error: The CSV file has fewer than 7 columns. Please check if EPU is pasted
in Column G.")
}

# 3. Data Pre-processing (Log-Difference)
# Establish date format and take log-differences to ensure stationarity
data_clean <- raw_data %>%
  mutate(
    # Establish Date format (set as the 1st day of each month)
    Date = as.Date(paste(Year, "-", Month, "-01")),
    # Log-transformation of variables
    ln_CO2 = log(as.numeric(CO2)),
    ln_Total = log(as.numeric(TOTAL)),
    ln_Dom = log(as.numeric(DOMESTIC)),
    ln_Intl = log(as.numeric(INTERNATIONAL)),
```

```
    ln_EPU = log(as.numeric(EPU))
  ) %>%
  na.omit() %>%
  mutate(
    # Calculate First Differences
    d_CO2 = c(NA, diff(ln_CO2)),
    d_Total = c(NA, diff(ln_Total)),
    d_Dom = c(NA, diff(ln_Dom)),
    d_Intl = c(NA, diff(ln_Intl)),
    d_EPU = c(NA, diff(ln_EPU))
  ) %>%
  na.omit() %>%
  arrange(Date)

# Prepare Wavelet Matrices (cbind: time index, variable value)
t <- 1:nrow(data_clean)
y_co2 <- cbind(t, data_clean$d_CO2)          # Dependent Variable: CO2
z_epu <- cbind(t, data_clean$d_EPU)          # Control Variable: EPU
x_total <- cbind(t, data_clean$d_Total)
x_dom <- cbind(t, data_clean$d_Dom)
x_intl <- cbind(t, data_clean$d_Intl)

# 4. Generate Table I: Descriptive Statistics and Unit Root Tests
# This section outputs Table 1 with multiple unit root tests (ADF, PP, KPSS)
calc_stats_revised <- function(x) {
  mean_val <- round(mean(x), 4)
  sd_val <- round(sd(x), 4)
  min_val <- round(min(x), 4)
  max_val <- round(max(x), 4)

  # Jarque-Bera Normality Test
  jb <- jarque.bera.test(x)
  jb_stat <- paste0(round(jb$statistic, 2), ifelse(jb$p.value < 0.01, "***",
ifelse(jb$p.value < 0.05, "**", "")))

  # ADF Test for Stationarity (H0: Unit root/Non-stationary)
  adf <- adf.test(x)
  adf_stat <- paste0(round(adf$statistic, 2), ifelse(adf$p.value < 0.01, "***",
```

```
ifelse(adf$p.value < 0.05, "**", "")))

  # PP Test for Stationarity (H0: Unit root/Non-stationary)
  pp <- pp.test(x)
  pp_stat <- paste0(round(pp$statistic, 2), ifelse(pp$p.value < 0.01, "***",
ifelse(pp$p.value < 0.05, "**", "")))

  # KPSS Test for Stationarity (H0: Stationary) - Note: Low p-value indicates non-
stationarity
  kpss_res <- kpss.test(x, null="Level")
  kpss_stat <- paste0(round(kpss_res$statistic, 3), ifelse(kpss_res$p.value < 0.05,
"*", ""))

  return(c(mean_val, sd_val, min_val, max_val, jb_stat, adf_stat, pp_stat, kpss_stat))
}

target_vars <- data_clean %>% select(d_CO2, d_Total, d_Dom, d_Intl, d_EPU)
stats_table_revised <- as.data.frame(t(apply(target_vars, 2, calc_stats_revised)))
colnames(stats_table_revised) <- c("Mean", "Std. Dev.", "Min", "Max", "Jarque-
Bera", "ADF", "PP", "KPSS")
rownames(stats_table_revised) <- c("ΔlnCO2", "ΔlnTOUR_Total",
"ΔlnTOUR_Dom", "ΔlnTOUR_Int", "ΔlnEPU")

# Export Table 1
write.csv(stats_table_revised, "Table1_Revised_Full_Tests.csv")
print(stats_table_revised)

# 5. Execute PWC Analysis and Visualization (Technical Parameters)
# Following reviewer suggestions to enhance replicability:
# 1. Mother Wavelet: Morlet wavelet (w0=6) for optimal time-frequency resolution.
# 2. Scale Resolution: 12 sub-octaves per octave (dj=1/12).
# 3. Frequency Range: Focus on 32-64 month band (long-term
tourism/macroeconomic cycles).
# 4. Cone of Influence (COI): Shaded edge regions indicate potential boundary
effects.

if(!require(fields)) install.packages("fields")
library(fields)
```

```
# Global Settings
par(family = "serif") # Use Times New Roman
my_palette <- tim.colors(64)
n_rands <- 200 # Note: Set to 1000 for final manuscript submission

# Time Axis Configuration
date_ticks <- seq(min(data_clean$Date), max(data_clean$Date), by = "3 years")
at_ticks <- match(date_ticks, data_clean$Date)
labels_ticks <- format(date_ticks, "%Y")

# Figure 1: Model 1 - Total Tourism
par(mfrow = c(1,1), mar = c(3,4,1,4))
pwtc_total <- pwtc(y_co2, x_total, z_epu, nrands = n_rands, dj = 1/12)

# Main Plotting
plot(pwtc_total,
     plot.phase = TRUE, # Display phase arrows for lead/lag relationship
     plot.cb = FALSE,
     xaxt = 'n',
     main = "", xlab = "", ylab = "", cex.axis = 1.1, lwd.coi = 1)
# Note: Areas outside the white semi-transparent line are the COI (statistical bias may occur)
axis(1, at = at_ticks, labels = labels_ticks, las = 1, family = "serif")

# Legend Plotting (Adjusted for proximity to main plot)
plt_coords <- par("plt")
image.plot(legend.only = TRUE, zlim = c(0, 1), col = my_palette, legend.width = 1.2,
           smallplot = c(0.89, 0.92, plt_coords[3], plt_coords[4]),
           axis.args = list(cex.axis = 0.8, family = "serif"))

# 6. Quantitative Analysis of Phase Angles and Lead-Lag Relationships
# Extract average phase angles for the 32-64 month band
calc_lead_lag <- function(pwc_obj, model_name) {
  scales <- pwc_obj$period
  idx_band <- which(scales > 32 & scales <= 64)
  avg_phase <- mean(pwc_obj$phase[idx_band, ], na.rm = TRUE)
```

```
  # Lead Month Formula: Lag = (Phase / (2*pi)) * Representative Period (48 months)
  lead_months <- (avg_phase / (2 * pi)) * 48

  cat("\n---", model_name, "---")
  cat("\nAvg Phase Angle (Radians):", round(avg_phase, 4))
  cat("\nAt 48-month cycle, Tourism leads CO2 by approx:", round(lead_months, 2),
"months")
  cat("\nInterpretation: Arrows pointing SE (0 to -pi/2) indicate Tourism leads
CO2\n")
}

calc_lead_lag(pwtc_total, "Model 1: Total Tourism")
calc_lead_lag(pwtc_dom, "Model 2: Domestic Tourism")
calc_lead_lag(pwtc_intl, "Model 3: International Tourism")

# 7. Quantitative Analysis of Structural Shifts (Post-Pandemic Recovery)
cat("\n--- Structural Change Analysis: International Tourism ---")
pre_idx <- which(data_clean$Date < as.Date("2020-01-01"))
post_idx <- which(data_clean$Date >= as.Date("2021-06-01"))
period_idx <- which(pwtc_intl$period >= 32 & pwtc_intl$period < 64)

r2_matrix <- pwtc_intl$rsq[period_idx, ]
avg_r2_pre <- mean(r2_matrix[, pre_idx], na.rm = TRUE)
avg_r2_post <- mean(r2_matrix[, post_idx], na.rm = TRUE)

# Statistical Test for R-squared differences
t_test_res <- t.test(as.vector(r2_matrix[, post_idx]), as.vector(r2_matrix[, pre_idx]))
cat("\nPre-pandemic R2:", round(avg_r2_pre, 4))
cat("\nPost-pandemic R2:", round(avg_r2_post, 4))
cat("\nCoherence Growth Rate:", round((avg_r2_post - avg_r2_pre) / avg_r2_pre *
100, 2), "%")
cat("\nP-value (t-test):", t_test_res$p.value)

if(t_test_res$p.value < 0.01) {
  cat("\nConclusion: Statistical evidence confirms a significant structural shift in the
Intl Tourism-CO2 relationship.")
}
```